MULTIMEDIA UNIVERSITY

STUDENT ID NO

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 1, 2018/2019

ECP4136 – JAVA TECHNOLOGY

(RE / TE)

23 OCTOBER 2018
9:00 A.M – 11:00 A.M.
(2 Hours)- OPEN BOOK

INSTRUCTIONS TO STUDENT

1. This examination paper consists of FIVE (5) pages including cover with THREE (3) questions only.

2. Answer ALL questions. The distribution of the marks for each question is given.

3. This is an Open-Book examination.

4. Please write all your answers in the Answer Booklet provided.

Instructions:
Figure 1 shows the UML diagram of `Shape` interface, `SquareOutline` class, `SquareSolid` class, `CircleOutline` class and `CircleSolid` class.

You are to "implement" the UML diagram in **Question 1** and "apply" the UML diagram in **Question 2** and **Question 3**.

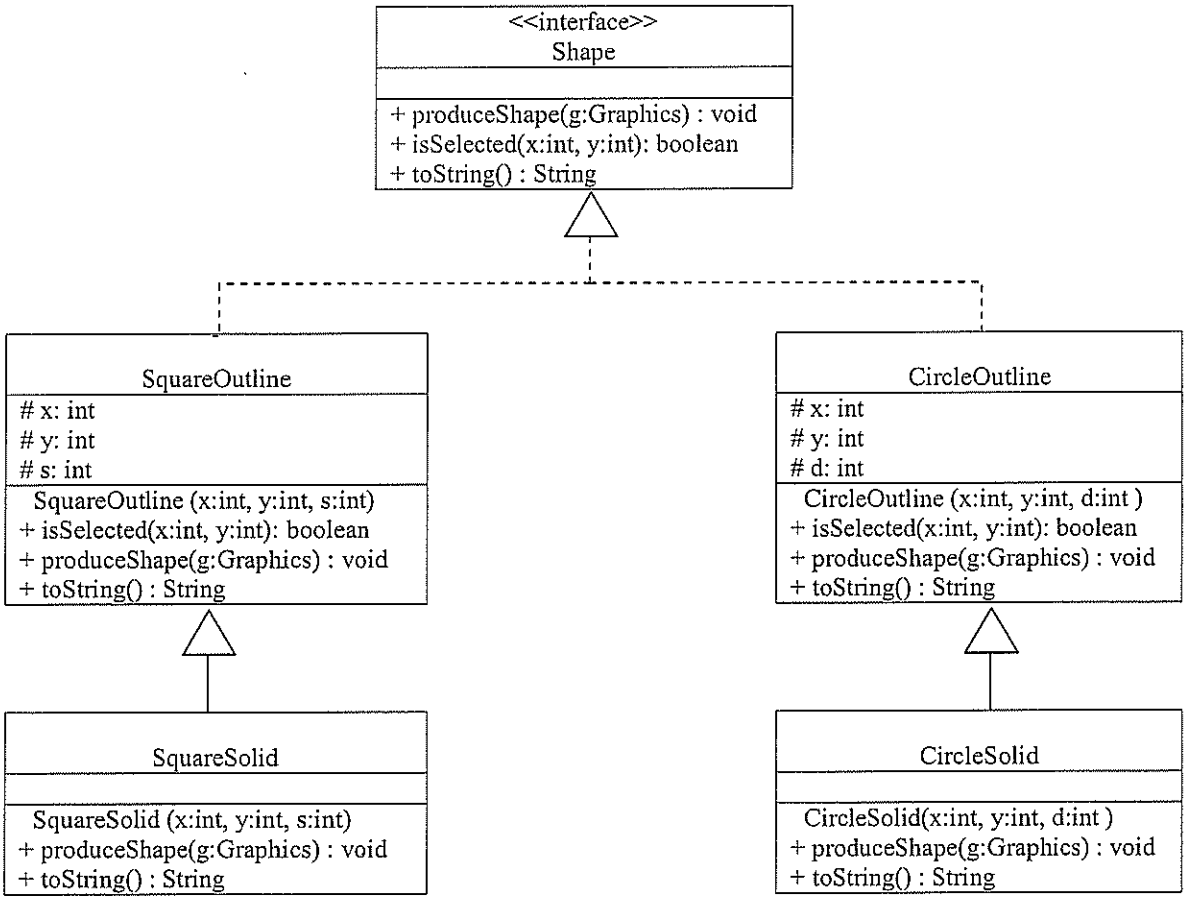

Figure 1

Continued...

Question 1

Implement the UML diagram shown in Figure 1. The marks distribution is as follows:

Create the `Shape` interface and its corresponding methods. [2 marks]

Implement the `SquareOutline` class with the following requirements:

- Declare of `x`, `y`, `s` variables. (`x,y`) is the upper left hand corner coordinate and `s` pixels is the size. [1 mark]
- Create constructor that will save all parameters in arguments. [1 marks]
- Implement `isSelected` method by checking if the given coordinate is within boundary of the square. [2 marks]
- Implement `produceShape` method by drawing the square. [1 mark]
- Implement `toString` method whereby if `x, y, s` is `100, 150, 50`, it will return "`drawSquare 100 150 50`". [1 mark]

Implement the `SquareSolid` class with the following requirements:

- Create constructor that will call superclass constructor. [1 marks]
- Implement `produceShape` method by drawing the filled square. [1 mark]
- Implement `toString` method whereby if `x, y, s` is `100, 150, 50`, it will return "`fillSquare 100 150 50`". [1 mark]

Implement the `CircleOutline` class with the following requirements:

- Declare `x`, `y`, `d` variables. (`x,y`) is the upper left hand corner coordinate and `d` pixels is the diameter. [1 mark]
- Create a constructor that will save all parameters in arguments. [1 marks]
- Implement `isSelected` method by checking if the given coordinate is within boundary of the square. [2 marks]
- Implement `produceShape` method by drawing the circle. [1 mark]
- Implement `toString` method whereby if `x, y, d` is `100, 150, 50`, it will return "`drawCircle 100 150 50`" [1 mark]

Implement the `CircleSolid` class with the following requirements:

- Create a constructor that will call superclass constructor. [1 marks]
- Implement `produceShape` method by drawing a filled circle. [1 mark]
- Implement `toString` method whereby if `x, y, d` is `100, 150, 50`, it will return "`fillCircle 100 150 50`". [1 mark]

Continued...

Question 2

Create a `ShapeIO` class that uses a text file to read and write a series of `Shape` object. **Sample** content of a file is shown in **Figure 2**. The class has following specifications:

Create a static `+write(list:ArrayList<Shape>, filename:String):void` method that throws `IOException`. The marks distribution is as follows:

- Correct declaration of method. [1 mark]
- Creation of `PrintWriter` object. [1 mark]
- Iteration of `list` and printing of the content from `Shape`'s `toString` method into the output file. [2 marks]
- Closing the `PrintWriter` object. [1 mark]

Create a static `+read(filename:String):ArrayList<Shape>` method that throws `IOException`. The marks distribution is as follows:

- Creation of a `Scanner` object using the filename provided. Create also an `ArrayList` container that will store `Shape` object. [2 marks]
- Using `Scanner` object to iterates through the files line by line, followed by word by word. [2 marks]
- Capturing the action word and the three parameters. [2 marks]
- Create appropriating `Shape` object and inserting it into `ArrayList` container. [4 marks]

```
drawCircle  50  50 50
drawSquare 100  50 50
drawCircle 150  50 50
drawSquare 200  50 50
fillSquare  50 100 50
fillCircle 100 100 50
fillSquare 150 100 50
fillCircle 200 100 50
drawCircle  50 150 50
drawSquare 100 150 50
drawCircle 150 150 50
drawSquare 200 150 50
fillSquare  50 200 50
fillCircle 100 200 50
fillSquare 150 200 50
fillCircle 200 200 50
```

Figure 2

Continued...

Question 3

Create the program shown in **Figure 3**. The program is used to visualize a series of drawing command from a text file. The program is implemented using a `VisualizerPanel` class. The specification of the class is as follows:

It has the following instance variables: [1 marks]

- `list:ArrayList<Shape>` object for holding a list of `Shape` object
- `index:int` that points to the current selected `Shape` in `list`. Default is `-1`;

The constructor that receives `ArrayList<Shape>` objects and performs the following action: [3 marks]

- Save a copy of the `Arraylist` argument
- Use the size (`300 x 300`) for the preferred size.
- Add mouse listener to listen to mouse press.

When mouse is pressed, the position of the mouse is used to determine which `Shape` object from `list` is selected. The `index` is updated and the panel will be repainted. [5 marks]

It has a `paintComponent` method that will iterate through the `list` object and call the `produceShape` method. Finally, `index` is used to highlight the selected `Shape` object using red color and the `toString()` value will be shown at position (0, 290). Note: The sample output is using the text file from Question 2. [6 marks]

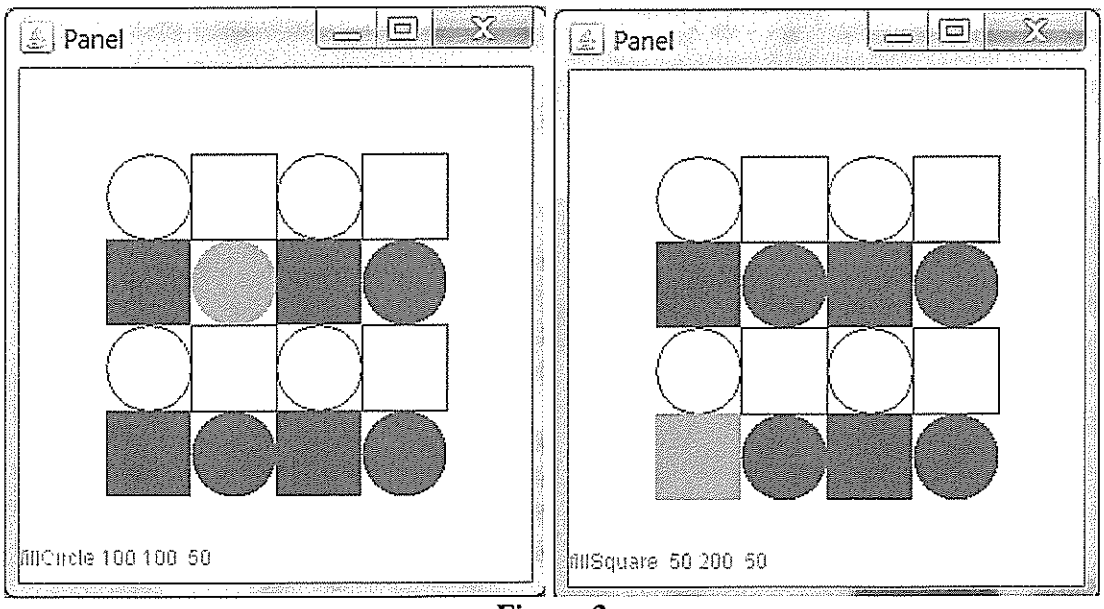

Figure 3

End of Paper